



TANGO Device Server

Generic Mirror Device
User's Guide

GenericMirror Class

Revision: release_1_3_2 - Author: langlois
Implemented in C++ - CVS repository: tango-ds

Introduction:

This device is used to control a Mirror:
when the user write on the theta attribute, the device calculate (via Optical Formulas) the values to send to the underlying devices (theta rotation,bender)

Class Identification:

- **Contact :** at synchrotron-soleil.fr - langlois
- **Class Family :** Motion
- **Platform :** All Platforms
- **Bus :** Not Applicable

Class Inheritance:

- Tango::Device_4Impl
  - GenericMirror

Class Description:

This device is used to pilot a Mirror

Properties:

Device Properties		
Property name	**Property type**	**Description**
AttributeCurvatureName	Tango::DEV_STRING	If a bender is used this property gives the name of the curvature attribute of the bender device.
AttributePositionName	Tango::DEV_STRING	Give the name of the attribute to move the motor according the motors device used.
AttributeThetaName	Tango::DEV_STRING	The name of the attribute to reach for the theta. For instance, if a TPP is used to rotate the mirror, the properties must be set to pitch
BenderDeviceName	Tango::DEV_STRING	If a bender is used this property gives the name of the bender device to reach
ChangeStripMotorName	Tango::DEV_STRING	This method allows to change the strip of the mirror
CommandStateName	Tango::DEV_STRING	Gives the name of the STATE command according the target device. For SimulatedMotor it is State.
CommandStopName	Tango::DEV_STRING	Name of the command stop for the reached device.
DistanceBetweenMirrorAndImageQ	Tango::DEV_DOUBLE	This is the q parameter representing the distance between the mirror and the image (infinity so INVALID for a focalisation mirror)
DistanceBetweenSourceAndMirrorP	Tango::DEV_DOUBLE	This is the p parameter representing the distance between the beam source and the mirror (infinity so INVALID for a collimation mirror)
HasBender	Tango::DEV_BOOLEAN	Indicate if there is a bender linked to the mirror
MirrorType	Tango::DEV_SHORT	The type of the mirror : - ’0’ for a normal mirror - ’1’ for a focusing mirror - ’2’ for a collimation mirror - ’3’ for an elliptic mirror
StripNumber	Tango::DEV_USHORT	The number of strip of the mirror. Set to 1 if there is no strip.

StripValues	Array of double	The values of the reference position of each mirror strip. These values are used to position the mirror to the wanted strip.
StripNames	Array of string	The name of the strip.
ThetaMotorName	Tango::DEV_STRING	The name of the motor which can move the theta mirror angle.
HasChangeStripMotor	Tango::DEV_BOOLEAN	This propertie is used to indicate if there is a motor to change strip.
CanRotate	Tango::DEV_BOOLEAN	This property indicate if the mirror have a motor to rotate. This property must be set to TRUE if there is a motor and FALSE if not.
RotationAngle	Tango::DEV_DOUBLE	If the CanRotate property is set to TRUE, then this property is used to set the default angle of the mirror.
BenderCurvatureUnit	Tango::DEV_USHORT	This property is used to indicate what is the unit of the bender curvature if the miror use a bender. Indeed, the curvature value is computed thanks the next equation : 1/R = (p+q)*sin(theta)/(2*p*q) But the bender device is waiting a new value of curvature in the same unit as it is configured in the device bender. So it is necessary to compute (in the generic mirror device) a good value of 1/R in the same unit as in the bender. 1 - bender curvature is in m-1 2 - bender curvature is in km-1
ThetaMotorUnit	Tango::DEV_USHORT	This property is used to make coincidate the unit of the theta (for instance pitch of a TPP in mrad) with the value of theta in the mirror device. The possible choice are : 1 - mrad 2 - degre
DefaultStrip	Tango::DEV_ULONG	The default strip index to be used when a init command is called
StripValuesTolerance	Tango::DEV_DOUBLE	Tolerance on the strip values. if the strip setting does not correspond to the motor position +/- the tolerance, the device will be disabled

Device Properties Default Values:

Property Name	Default Values
AttributeCurvatureName	No default value
AttributePositionName	position
AttributeThetaName	No default value
BenderDeviceName	No default value
ChangeStripMotorName	No default value
CommandStateName	state
CommandStopName	stop
DistanceBetweenMirrorAndImageQ	No default value
DistanceBetweenSourceAndMirrorP	No default value
HasBender	No default value
MirrorType	No default value
StripNumber	No default value
StripValues	No default value
StripNames	No default value
ThetaMotorName	No default value
HasChangeStripMotor	false
CanRotate	No default value
RotationAngle	No default value
BenderCurvatureUnit	No default value
ThetaMotorUnit	1
DefaultStrip	1
StripValuesTolerance	0

There is no Class properties.

States:

States	
Names	**Descriptions**
MOVING	Mirror is MOVING (at least one motor is Moving)
STANDBY	Mirror is STANDBY (all motors are Standby)
ALARM	Mirror is ALARM (at least one motor is Alarm)
FAULT	Mirror is FAULT not possible to reach a device, getting a value or a state. (In this case it is only allowed to redone a init after correcting the problem with the other devices)
INIT	The device is in this state after the Init command succeeded. After it needs a InitializeMirror call to be fully use.
DISABLE	The strip does not correspond to the motor position.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
theta: The angle of the mirror	DEV_DOUBLE	READ_WRITE	No
pDistance: The distance between source and mirror in meter	DEV_DOUBLE	READ	No
qDistance: The distance between mirror and image in meter	DEV_DOUBLE	READ_WRITE	No
type: The type of the mirror : - '0' for a normal mirror - '1' for a focusing mirror - '2' for a collimation mirror - '3' for an elliptic mirror	DEV_STRING	READ	No
currentStrip: The current strip index	DEV_USHORT	READ	No
currentStripName: The name of the strip	DEV_STRING	READ	No
curvature: The value of the curvature of the bender device (if in use)	DEV_DOUBLE	READ	No
tx: The translation value to switch between mirror strips	DEV_DOUBLE	READ	Yes
isBenderLess: If IsBenderLess attribute is notched, only the mirror can move. The bender doesn't received any value according a new angle	DEV_BOOLEAN	READ_WRITE	Yes

Commands:

More Details on commands....

Device Commands for Operator Level		
Command name	**Argument In**	**Argument Out**
Init	DEV_VOID	DEV_VOID
State	DEV_VOID	DEV_STATE
Status	DEV_VOID	CONST_DEV_STRING
Stop	DEV_VOID	DEV_VOID
InitializeMirror	DEV_VOID	DEV_VOID
ChangeStrip	DEV_USHORT	DEV_VOID

1 - Init

- **Description:** This commands re-initialise a device keeping the same network connection.
  After an Init command executed on a device, it is not necessary for client to re-connect to the device.
  This command first calls the device *delete_device()* method and then execute its *init_device()* method.
  For C++ device server, all the memory allocated in the *nit_device()* method must be freed in the *delete_device()* method.
  The language device desctructor automatically calls the *delete_device()* method.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  DEV_VOID : none.

- **Command allowed for:**
  - Tango::MOVING
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::FAULT
  - Tango::INIT
  - Tango::DISABLE

2 - State

- **Description:** This command gets the device state (stored in its *device_state* data member) and returns it to the caller.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  DEV_STATE : State Code

- **Command allowed for:**
  - Tango::MOVING
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::FAULT
  - Tango::INIT
  - Tango::DISABLE

3 - Status

- **Description:** This command gets the device status (stored in its *device_status* data member) and returns it to the caller.

- **Argin:**
  DEV_VOID : none.

- **Argout:**
  CONST_DEV_STRING : Status description

- **Command allowed for:**
  - Tango::MOVING
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::FAULT
  - Tango::INIT
  - Tango::DISABLE

4 - Stop

- **Description:** Stop all the motors

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::MOVING
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::DISABLE

5 - InitializeMirror

- **Description:** Method to initialize all the object needed by the device mirror

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::INIT
  - Tango::DISABLE

6 - ChangeStrip

- **Description:** Method to change the wanted strip of the mirror

- **Argin:**
  DEV_USHORT : The index of the strip

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::STANDBY
  - Tango::ALARM
  - Tango::DISABLE